IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.: 09/980,271 Filed: November 30, 2001 Inventor(s): Alain Delpuch	§	Examiner: Brown, Rueben M. Group/Art Unit: 2623 Atty. Dkt. No: 5266-09100
Title: Flexible Interface For Secure Input of Pin Code	§	****CERTIFICATE OF E-FILING TRANSMISSION**** I hereby certify that this correspondence is being transmitted via electronic filing to the United States Patent and Trademark Office on the date shown below Rory D. Rankin (Printed Name) / Rory D. Rankin / (Signature) November 7, 2007 (Date)

REPLY BRIEF TO EXAMINER'S ANSWER

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir/Madam:

This Reply Brief is filed in Response to the Examiner's Answer mailed on September 7, 2007.

REMARKS

Claims 1-11 are pending. Claims 1-11 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,338,138 (hereinafter "Raduchel"), in view of U.S. Patent No. 6,122,741 (hereinafter "Patterson").

The below is presented in response to the Examiner's Response to Argument in the Examiner's Answer. Appellants have presented below responses to any *new* statements by the Examiner. Thus, where no new discernable arguments were presented in the Examiner's Answer, Appellants have not added further responses, but relied on those presented in the Appeal Brief.

The Examiner's Response to Applicant's Argument is itemized as section 10 on page 7 of the Examiner's Answer. Appellants will refer to paragraphs of section 10 in the discussion below as appropriate.

In the first paragraph of section 10, the Answer states:

> "Examiner notes that the password discussed in Raduchel, corresponds with the PIN as claimed."

Applicant agrees that this is the examiner's position. Applicant further notes, as discussed in the Appeal Brief, that every occurrence of the "password" in Raduchel refers to a logon procedure whereby a use logs on to a computer system. For example, Raduchel discloses:

> "In accordance with methods and systems consistent with the present invention, when a stand-alone computer starts up, the user is unable to utilize any of the services of the computer until an authentication process is successfully completed. To perform authentication, the computer runs a browser with an applet that displays a logon screen to the user, who then enters log-in information (e.g., username and password)." (Raduchel, col. 3, lines 22-29).

"Internet 102 contains security node 118 with CPU 120, secondary storage device 122, memory 124, and at least one I/O device 126. Secondary storage device 122 contains authentication file 130, storing the data against which users are authenticated . . . Authentication file 130 contains the user name and password for authenticated users." (Raduchel, col. 4, lines 10-16).

"FIG. 2, which depicts a flowchart of the steps performed at start-up time of local computer 101. When the local computer is initially started, a small portion of the operating system is loaded (step 202). In this step, the minimum code necessary to run authentication is loaded, including VM 117 as well as the minimum components of the operating system necessary to load and run a web browser; it does not include a command interpreter or file capabilities.

Next, the browser is loaded and run . . . when running the browser, the user is initially presented with a screen 300 having a login dialog box 302 into which the user can enter their username and password. This screen is displayed by an applet, stored with the browser, that performs authentication by communicating with the authentication manager. . . . However received, the authentication information, including the username and password, is sent by the browser to the authentication manager . . . .

The authentication manager receives the log-in information . . . containing a user name and password (step 402 in FIG. 4). After receiving this information, the authentication manager authenticates the user by accessing the authentication file to determine if the user name and password are contained in it (step 404) and returns a token that identifies the services that the user may use (step 406). Additionally, this token may contain a profile of the user's access rights, and when the token is returned to the local computer, it would be included in all further requests from the local computer." (Raduchel, col. 4, line 59 – col. 3, line 31).

Therefore, every reference in Raduchel to the "password" refers to the log on procedure whereby a user is authenticated. Note that prior to authentication the loaded portions of the operating system are only those necessary to perform authentication. The browser is used as part of the authentication (i.e., provides a interface for the user to enter a username and password). It is noted that the portions of the operating system which are

loaded at this time do not include a command interpreter or file capabilities. It is during this authentication that all access rights of the user are determined. A token identifying the users rights is then returned to the local computer – and this token is used for all further requests from the computer.

In the third paragraph, the Answer states:

> "First of all, it is noted that while the claims do recite that, 'crypted information corresponding to PIN code entering key-pressing operations are received from the security', this step is at the back-end of the authentication process. The front-end of the authentication process is the claimed, `security manager configured to receive a request for user authentication from the application'. However, the claims do not explicitly point out where in the system the `security manager' operates. This is important, because the claims as written are broad enough to read on the security manger operating at the server or database."

Applicant agrees the presently claimed invention is written such that it may read on the security manager operating at the server or database. This does not affect Applicant's analysis provided in the Appeal Brief and the claims remain patentably distinct from the cited art even given such an interpretation.

In paragraphs 4-5 of the Answer it states:

> "Secondly, it is noted that the claims do not recite any entity that receives the PIN (on the front end), i.e., directly from the input device and delivers the PIN to the claimed `security manager'. Thus the claims are broad enough to read on the security manger receiving the PIN code from some other entity, (including an intermediary or the application itself) and then comparing the received PIN code, with the stored value, as recited.
>
> Even though Appellant has argued throughout that PIN code is hidden from the application, there is no language recited in claims that support such a requirement."

Applicant does not agree with the above statements. Claim 1 recites a system which includes the recitation "wherein the entered PIN code is not supplied to the application." Further, even were one to accept the examiner's arguments in paragraphs 4-5, the rejections remain improper.

In paragraphs 8-10, the Answer states the following:

> "Appellant argues on page 13 that the browser in Raduchel is separate from the application that user desires authentication for, and since it is the browser that presents the dialog box, instead of the application, Raduchel does not meet the claim. Examiner notes that Raduchel more specifically teaches that the first time that the user selects an icon corresponding to a particular application that the browser sends a request to the authentication manager, which then sends the appropriate applet that will display the logon screen, col. 3, lines 5-28 & col. 5, lines 5-8. Subsequently, the applet is already stored at the PC and is called on by the browser, when the user again requests to be authenticated to the particular application, col. 5, lines 50-62."

However, this characterization of Raduchel above is incorrect. In contrast to that suggested above, icons are only displayed for services that has user has rights to as determined during a log on authentication procedure. Icons are presented after log on and authentication has been performed. Selecting an icon for a particular application does not result in the downloading of an applet from the authentication manager and display of the logon screen. As discussed above, all references to the cited password in Raduchel refer to a log on authentication procedure at start up. This authentication procedure determines the user's rights and returns a token which outlines those rights. This token is used for all future requests. Services for which a user has been authenticated may then depicted by corresponding icons. A user may then select one of these icons for access to a given service. If a copy of the program code corresponding to a selected icon is not already present on the local computer, it is downloaded. However, there is no further or additional authentication procedure performed by selecting the icon. The user's rights have been determined during the initial authentication procedure and the icons (or lack

thereof) represent the user's access rights. Raduchel makes this clear as disclosed in the following:

> "FIG. 2, which depicts a flowchart of the steps performed at start-up time of local computer 101. When the local computer is initially started, a small portion of the operating system is loaded (step 202). In this step, the minimum code necessary to run authentication is loaded, . . .
> Next, the browser is loaded and run . . . when running the browser, the user is initially presented with a screen 300 having a login dialog box 302 into which the user can enter their username and password. This screen is displayed by an applet, stored with the browser, that performs authentication by communicating with the authentication manager" (Raduchel, col. 4, line 59 – col. 5, line 7).

> "The authentication manager receives login information from the local computer, verifies this information against an authentication file, and returns indications of the services on the local computer that the user is able to utilize. The local computer receives these indications and displays icons representing the services available to that user. The user may then select an icon, causing an applet to be downloaded from the authentication manager (or another server) onto the local computer to facilitate the user's utilization of the corresponding service." (Raduchel, Summary of Invention, col. 2, lines 16-25).

> "In accordance with methods and systems consistent with the present invention, when a stand-alone computer starts up, the user is unable to utilize any of the services of the computer until an authentication process is successfully completed. To perform authentication, the computer runs a browser with an applet that displays a logon screen to the user, who then enters log-in information (e.g., username and password). Upon receiving this log-in information, the applet transfers it to an authentication manager, remotely located somewhere in the network, that determines whether the user should be able to use all the available services of the computer or only a limited subset of the available services. If the user is authenticated, the authentication manager enables the user to use additional services of the computer, such as access to files, change calendar information, and access to applications that the user is otherwise authorized to use. To do so, the authentication manager downloads to the browser an indication of the services the user is able to use. The browser then displays icons indicating each of these services, and the user may select the icons, causing applets that either perform these services or provide access to these services to be downloaded to the browser and run, thus enabling the user to utilize the services.

If the user is not authenticated, the authentication manager enables him to only utilize a subset of the services provided by the computer, such as calendaring and e-mail; he is unable to utilize other services provided by the computer such as accessing the local file system. To enable the user to use this subset, the authentication manager downloads an indication of the services the user is allowed to use, and the browser displays icons which, when selected, cause applets to be downloaded that facilitate use of these services." Raduchel, col. 3, lines 22-54).

"If authentication fails, the browser provides the user with restricted access to the local computer (step 210). In this step, the browser displays icons representative of the services that the user may use, as indicated in the token received from the authentication manager. . . . Upon selecting one of the icons 502-506 for the first time, the browser sends a request to the authentication manager for the appropriate service applet, and the authentication manager downloads it to the browser so that the user may use the corresponding service. Subsequent selections of the icon do not cause a download of the service applet; instead, recognizing that a copy has already been downloaded, the browser merely invokes that copy." (Raduchel, col. 5, lines 46-62).

"In the situation where the services are provided by the operating system, the user is unable to utilize the services until authentication is successfully completed and the user is granted access to those services. To gain access to one of these services, code must be downloaded from the authentication manager, in the form of an applet, that provides a user interface to the service. For example, if the service is a file system, the applet provides a command line or other graphical user interface so that the user could enter commands to manipulate the file system. An "applet" comprises code that usually runs in another program like a browser. In the situation where the services are remote, the user is unable to utilize the services because the code that performs the services are applets downloaded from the authentication manager (or other server) only after successful authentication. In accordance with methods and systems consistent with the present invention, when a stand-alone computer starts up, the user is unable to utilize any of the services of the computer until an authentication process is successfully completed. To perform authentication, the computer runs a browser with an applet that displays a logon screen to the user, who then enters log-in information (e.g., username and password)."

Accordingly, Raduchel clearly discloses authentication is performed at the time of startup/logon. At that time the user's access rights are determined by the log on authentication procedure. The services to which the user has rights may then be depicted as icons. Each of these icons represent services for which the user has already been authenticated. The first time one of these icons is selected, corresponding code may be downloaded if a copy is not already present on the local computer. Nowhere does Raduchel disclose "that the first time that the user selects an icon corresponding to a particular application that the browser sends a request to the authentication manager, which then sends the appropriate applet that will display the logon screen" as suggested by the examiner's Answer. The user's rights to the service represented by the icon have already been determined. Therefore, as discussed in the Appeal Brief, nowhere does Raduchel disclose the connection between an application to which the user seeks access and a corresponding authentication procedure as recited in the claim.

Raduchel simply discloses a log on authentication procedure.

CONCLUSION

For the foregoing reasons, it is submitted that the remaining rejections are erroneous, and reversal of the rejections is respectfully requested.

The Commissioner is authorized to charge any fees that may be due to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5266-09100/RDR.

Respectfully submitted,

/Rory D. Rankin/

Rory D. Rankin
Reg. No. 47,884
Attorney for Appellant

Meyertons, Hood, Kivlin, Kowert & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
(512) 853-8866

Date: November 7, 2007